# Spam Email Classifier Project - Documentation

## 1. Project Overview

This project is a complete spam email classifier using FastAPI for the backend and HTML/Tailwind for the frontend. It uses a Naive Bayes classifier trained on TF-IDF features to classify emails as spam or ham.

## 2. Project Structure

spam\_classifier/  
├── app/  
│ ├── \_\_init\_\_.py  
│ ├── main.py  
│ ├── config.py  
│ ├── schema.py  
│ ├── core/  
│ │ └── model\_handler.py  
│ ├── router/  
│ │ └── predict\_router.py  
│ └── utils/  
│ └── text\_cleaner.py  
├── templates/  
│ └── index.html  
├── static/  
├── saved\_models/  
│ ├── spam\_classifier.pkl  
│ └── tfidf\_vectorizer.pkl  
├── train\_model.py  
├── requirements.txt

## 3. Backend Components

• FastAPI is used to build a REST API.

• Naive Bayes model and TF-IDF vectorizer are trained using scikit-learn.

• TextCleaner class processes and cleans input text.

## 4. Frontend

• Simple frontend using HTML and TailwindCSS.

• Text area to input email text and a button to call API.

• Displays spam/ham prediction in UI.

## 5. Model Training Script

• Uses a sample dataset of spam/ham messages.

• Applies text cleaning, TF-IDF transformation, and trains Naive Bayes model.

• Saves model and vectorizer to saved\_models/ folder.

## 6. Running the Project

1. Install dependencies from requirements.txt

2. Run train\_model.py to generate model files

3. Start FastAPI server using: uvicorn app.main:app --reload

4. Visit http://localhost:8000 to use the frontend

## 7. requirements.txt

fastapi  
uvicorn  
scikit-learn  
joblib  
nltk  
jinja2